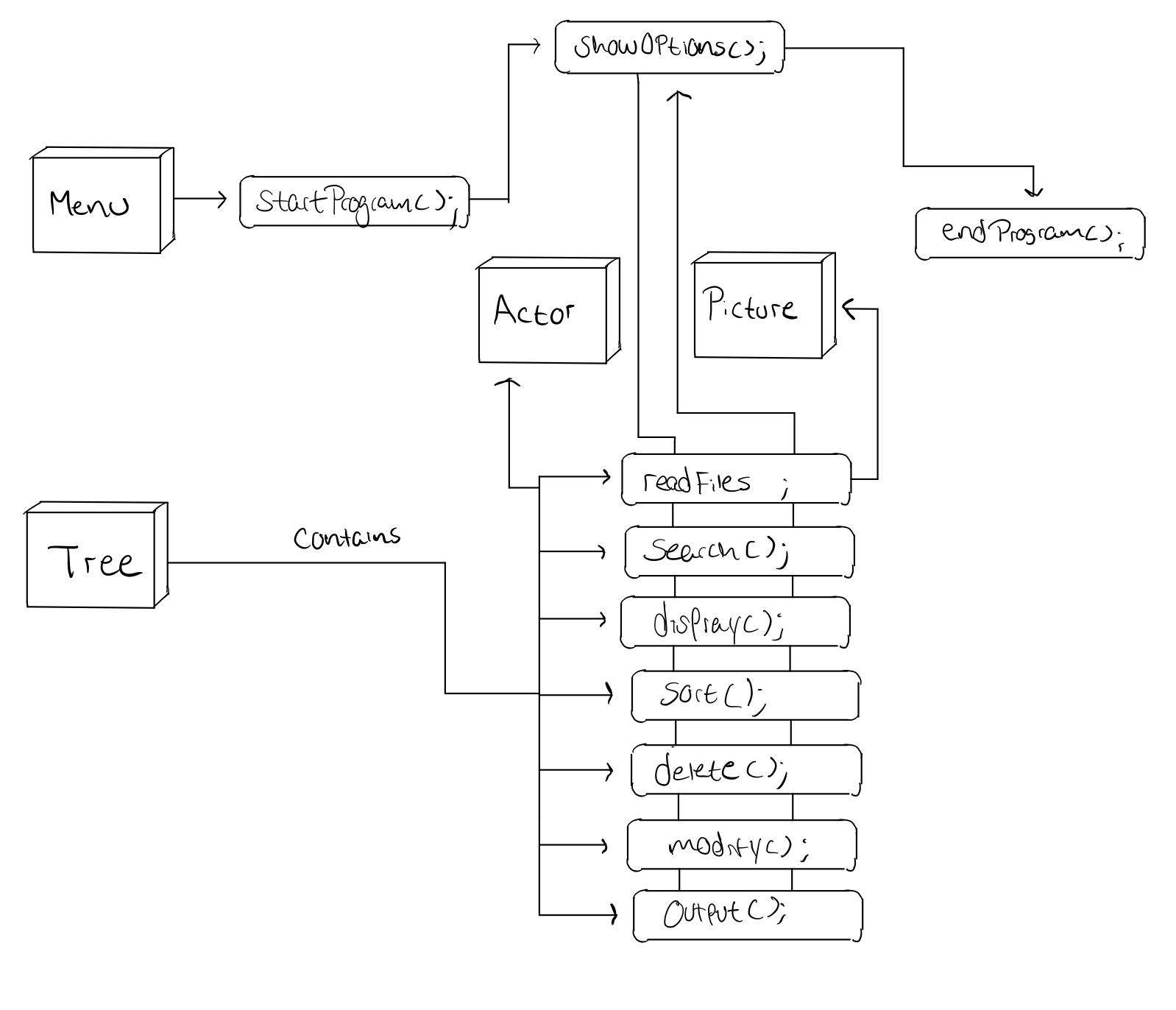
**Project Design/Implementation Document**

1. Title

* Database program
* Abyel Romero
* ID: 107066046
* Due: November 30, 2017

1. Problem DescriptionThe program is a simple database system. It will read and modify files full of records from the last century about movies and Oscar awards.
2. Overall Software ArchitectureThe program consists of two binary search tree classes, a menu class, an error class, and an actor and picture class. One tree is for the actor class and the other tree is for the picture class. So, I’ll have a tree for actors and a tree for pictures (the two files to be read). Each tree has their specific functions -to delete, search, display, sort, etc.- the different records that are on the tree. (after reading them). The menu class has simple functions to start the program (show a greeting), show options for the user to choose what to do, and end the program. I also thought having an error class was a good idea because it will allow me to have organized functions for the different exception handling in the program.

Menu

* + Calls the start program function
  + After the start program, it calls the show options function.
  + The show options functions show the different options for the user (display, search, sort, etc.) These are implemented in the tree object.
  + After the user selects a function, the program goes back to the show options functions and repeats the circle.
  + The user can end the program in here.

Tree

* + Has the different implementations of the functions for the trees

Actor

* + Object with the parameters for each actor

Picture

* + Object with the parameters for each picture (movie)

1. Input RequirementsThe program is to read two files: actor-actress.csv and pictures.csv. The first file contains some films since 1928 and their respective actor or actress nominations for the Oscar. It provides the type of award that the actor or actress was to receive and a Boolean variable if they received it or not. The second file contains more detailed aspects of the movies since 1927. It lists their ratings, two genres, release month, Metacritic score and a synopsis.
2. Output RequirementsThe program will provide two types of outputs: reports and updated Actor-Actress/Film information. The reports consist on the sorted fields of both files. The user chooses which fields to report. It would be in ASCIII format. The updated Actor-Actress/Film information is after the actions of the user (sorts, searches, etc.).

1. Problem Solution Discussion //I wasn’t very sure of what was asked in here.  
   For the main part of the program, the parameters are going to be read into an actor or picture object and the object will be put in a node of the binary search tree. The two trees contain nodes of those objects. The error class contains functions that I will call depending on which exception my program catches.
2. Data StructuresI believe implementing a hash table to search for something other than movie or actor name is a good idea. The efficiency of hash tables seems to go well in this database since it has a big number of records. Also, I feel hashes are used very often in the work space so it’ll do me good to learn more about them. I chose this over a vector because I understand vectors very well at this point and they are less efficient to search for something. I was thinking for linked lists, but they are also not as efficient and more complicated to implement than a vector. And another binary search tree was also a good idea but then again, learning hashes will be great.
3. User Interface SchemeThe main menu will provide options for the different trees. After the user selects and performs one of these options, there will be a function to exit the program.
4. Status of Application  
   The project will be developed on Xcode.